UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/737,129	12/14/2000	Sandra Johnson Baylor	YOR9-2000-0601US1(8728-42	9558

7590 04/30/2004

Frank Chau, Esq.
F. CHAU & ASSOCIATES, LLP
Suite 501
1900 Hempstead Turnpike
East Meadow, NY 11554

EXAMINER	
ALI, SYED J	
ART UNIT	**PAPER NUMBER**
2127	4

DATE MAILED: 04/30/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/737,129	BAYLOR ET AL.	
	Examiner	Art Unit	
	Syed J Ali	2127	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 December 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-33* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-33* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 December 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-33 are pending in this application.

Claim Objections

2. Claims 15 and 32 objected to because of the following informalities:

a. In line 3 of claim 15, "by machine" should read "by a machine".

b. In line 3 of claim 32, "by machine" should read "by a machine".

Appropriate correction is required.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. **Claims 1-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Torii (USPN 6,122,712) in view of Steckermeier et al. ("Using Locality Information in Userlevel Scheduling") (hereinafter Steckermeier).**

5. As per claim 1, Torii teaches the invention substantially as claimed, including a method for scheduling threads in a multi-processor computer system having an operating system and at least one cache, comprising the steps of:

storing in a first data structure thread ids for at least some of the threads associated with a context switch performed by the operating system, each of the thread ids uniquely identifying one of the threads (col. 3 lines 47-56); and

storing in a second data structure a plurality of entries for a plurality of groups of contiguous cache lines, each of the plurality of entries arranged such that a thread id in the first data structure is capable of being associated with at least one of the contiguous cache lines in at least one of the plurality of groups of contiguous cache lines, the thread identified by the thread id having accessed the at least one of the contiguous cache lines in the at least one of the plurality of groups of contiguous cache lines (col. 8 line 36 - col. 9 line 32).

6. Steckermeier teaches the invention as claimed, including the following limitations not shown by Torii:

mining for patterns in the plurality of entries in the second data structure to locate multiples of a same thread id that repeat with respect to at least two of the plurality of groups of contiguous cache lines (§3.2, §3.2.2); and

scheduling on a same processing unit the threads identified by the located multiples of the same thread id and any other threads identified by any other thread ids associated with the at least two of the plurality of groups of contiguous cache lines (§3.2).

7. It would have been obvious to one of ordinary skill in the art to combine Torii and Steckermeier since both references consider the problem of scheduling a thread on a processor such that the data the thread accesses is in the cache. However, Torii only teaches a data structure that represents the thread's locality information, while Steckermeier only teaches the scheduling algorithms used for placing the threads on processors. Thus, both are absent a

complete representation of how locality of a thread's data may be used to most efficiently schedule that thread. Specifically, while Torii teaches a method of representing a thread's access patterns according to a thread id and its associated cache line, the scheduling algorithm is left up to the operating system or programmer. Furthermore, Steckermeier teaches that threads that operate on the same data should be scheduled on the same processor, yet fails to specifically set forth how such information may be represented in the system. The combination of Torii and Steckermeier provide a complete model of how to represent data patterns and utilize them to reduce cache misses and increase the performance of a system.

8. As per claim 2, Torii teaches the invention as claimed, including the method according to claim 1, further comprising the step of adding and removing a group to the plurality of groups of contiguous cache lines when a contiguous cache line in the group is accessed by a given thread and when all contiguous cache lines in the group are flushed, respectively (col. 4 line 66 - col. 5 line 7; col. 13 lines 16-23).

9. As per claim 3, Torii teaches the invention as claimed, including the method according to claim 1, further comprising the step of restricting the plurality of groups to a finite number of groups (col. 4 lines 31-34).

10. As per claim 4, Steckermeier teaches the invention as claimed, including the method according to claim 3, further comprising the step of determining when there exists the finite number of groups (§3.4).

11. As per claim 5, Steckermeier teaches the invention as claimed, including the method according to claim 3, wherein said mining step is performed when there exists the finite number of groups (§3.4).

12. As per claim 6, Steckermeier teaches the invention as claimed, including the method according to claim 1, wherein said mining step is performed upon receipt of a command (§3.2.2).

13. As per claim 7, Steckermeier teaches the invention as claimed, including the method according to claim 1, wherein said mining step is performed at least of continuously, at predefined intervals, and upon an occurrence of at least one predefined event (§3.2.2).

14. As per claim 8, Steckermeier teaches the invention as claimed, including the method according to claim 1, wherein said mining step is performed in at least one of software and hardware (§3.2).

15. As per claim 9, Torii teaches the invention as claimed, including the method according to claim 1, wherein said second data structure is comprised of a plurality of rows and a plurality of columns (Fig. 16).

16. As per claim 10, Torii teaches the invention as claimed, including the method according to claim 9, wherein each of the plurality of groups of contiguous cache lines corresponds to one of the plurality of rows (Fig. 16).

17. As per claim 11, Torii teaches the invention as claimed, including the method according to claim 9, wherein each of the thread ids in the second data structure corresponds to one of the plurality of columns (Fig. 16, element 39).

18. As per claim 12, Torii teaches the invention as claimed, including the method according to claim 11, wherein each of the plurality of groups of contiguous cache lines corresponds to one of the plurality of rows and the any other threads correspond to any of the plurality of columns that intersect any of the plurality of rows corresponding to the at least two of the plurality of groups (Fig. 16, element 39).

19. As per claim 13, Torii teaches the invention as claimed, including the method according to claim 9, further comprising the step of allocating each of the plurality of rows to one of the plurality of groups of contiguous cache lines (Fig. 16, element 18).

20. As per claim 14, Torii teaches the invention as claimed, including the method according to claim 10, further comprising the step of, for each of a cache line in a group in the plurality of groups of contiguous cache lines, storing an index of a row corresponding to the group containing the cache line in the cache line (Fig. 16).

21. As per claim 15, Torii teaches the invention as claimed, including the method according to claim 1, wherein said method is implemented by a program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform said method steps (col. 3 lines 25-34).

22. As per claim 16, Torii teaches the invention substantially as claimed, including a method for scheduling threads in a multi-processor computer system having an operating system and at least one cache, comprising the steps of:

storing in a first data structure thread ids for at least some of the threads associated with a context switch performed by the operating system, each of the thread ids uniquely identifying one of the threads (col. 3 lines 47-56); and

storing in a second data structure a plurality of entries for a plurality of groups of contiguous cache lines, each of the plurality of entries arranged such that a thread id in the first data structure is capable of being associated with at least one of the contiguous cache lines in at least one of the plurality of groups of contiguous cache lines, the thread identified by the thread id having accessed the at least one of the contiguous cache lines in the at least one of the plurality of groups of contiguous cache lines (col. 8 line 36 - col. 9 line 32).

23. Steckermeier teaches the invention as claimed, including the following limitations not shown by Torii:

mining for patterns in the plurality of entries in the second data structure to locate multiples of a same thread id that repeat with respect to at least two of the plurality of groups of contiguous cache lines (§3.2, §3.2.2); and

mapping the threads identified by the located multiples of the same thread id to at least one native thread (§3.2.2).

24. As per claim 17, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein the threads identified by the located multiples of the same thread comprise m threads and the at least one native thread comprises n threads, m and n being integers, m being greater than n (§3.2).

25. As per claim 18, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein said method further comprises the step of scheduling on a same processing unit the threads identified by the located multiples of the same thread id and any other threads identified by any other thread ids associated with the at least two of the plurality of groups of contiguous cache lines (§3.2).

26. As per claim 19, Torii teaches the invention as claimed, including the method according to claim 16, further comprising the step of adding and removing a group to the plurality of groups of contiguous cache lines when a contiguous cache line in the group is accessed by a given thread and when all contiguous cache lines in the group are flushed, respectively (col. 4 line 66 - col. 5 line 7; col. 13 lines 16-23).

27. As per claim 20, Torii teaches the invention as claimed, including the method according to claim 16, further comprising the step of restricting the plurality of groups to a finite number of groups (col. 4 lines 31-34).

28. As per claim 21, Steckermeier teaches the invention as claimed, including the method according to claim 16, The method according to claim 16, further comprising the step of determining when there exists the finite number of groups (§3.4).

29. As per claim 22, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein said mining step is performed when there exists the finite number of groups (§3.4).

30. As per claim 23, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein said mining step is performed upon a receipt of a command (§3.2.2).

31. As per claim 24, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein said mining step is performed at least one of continuously, at predefined intervals, and upon an occurrence of at least one predefined event (§3.2.2).

32. As per claim 25, Steckermeier teaches the invention as claimed, including the method according to claim 16, wherein said mining step is performed in at least one of software and hardware (§3.2).

33. As per claim 26, Torii teaches the invention as claimed, including the method according to claim 16, wherein said second data structure is comprised of a plurality of rows and a plurality of columns (Fig. 16).

34. As per claim 27, Torii teaches the invention as claimed, including the method according to claim 26, wherein each of the plurality of groups of contiguous cache lines corresponds to one of the plurality of rows (Fig. 16).

35. As per claim 28, Torii teaches the invention as claimed, including the method according to claim 26, wherein each of the thread ids in the second data structure corresponds to one of the plurality of columns (Fig. 16, element 39).

36. As per claim 29, Torii teaches the invention as claimed, including the method according to claim 28, wherein each of the plurality of groups of contiguous cache lines corresponds to one of the plurality of rows and the any other threads correspond to any of the plurality of columns that intersect any of the plurality of rows corresponding to the at least two of the plurality of groups (Fig. 16, element 39).

37. As per claim 30, Torii teaches the invention as claimed, including the method according to claim 27, further comprising the step of allocating each of the plurality of rows to one of the plurality of groups of contiguous cache lines (Fig. 16, element 18).

38. As per claim 31, Torii teaches the invention as claimed, including the method according to claim 27, further comprising the step of, for each of a cache line in a group in the plurality of groups of contiguous cache lines, storing an index of a row corresponding to the group containing the cache line in the cache line (Fig. 16).

39. As per claim 32, Torii teaches the invention as claimed, including the method according to claim 16, wherein said method is implemented by a program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform said method steps (col. 3 lines 25-34).

40. As per claim 33, Torii teaches the invention substantially as claimed, including a method for scheduling threads in a multi-processor computer system having an operating system and at least one cache, comprising the steps of:

storing in a first data structure thread ids for at least some of the threads associated with a context switch performed by the operating system, each of the thread ids uniquely identifying one of the threads (col. 3 lines 47-56); and

storing in a second data structure a plurality of entries for a plurality of groups of contiguous cache lines, each of the plurality of entries arranged such that a thread id in the first

data structure is capable of being associated with at least one of the contiguous cache lines in at least one of the plurality of groups of contiguous cache lines, the thread identified by the thread id having accessed the at least one of the contiguous cache lines in the at least one of the plurality of groups of contiguous cache lines (col. 8 line 36 - col. 9 line 32).

41. Steckermeier teaches the invention as claimed, including the following limitations not shown by Torii:

identifying pools of threads in the plurality of entries in the second data structure such that each of the pools of treads comprises the threads identified by a same thread id that forms a multiple with respect to one of the plurality of groups of contiguous cache lines, the multiples repeating with respect to at least two of the plurality of groups of contiguous cache lines (§3.2, §3.2.2); and

scheduling on a same processing unit the threads identified by the located multiples of the same thread id and any other threads identified by any other thread ids associated with the at least two of the plurality of groups of contiguous cache lines (§3.2.2).

Conclusion

42. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Edler et al. (USPN 5,724,586; "Thread Scheduling for Cache Locality") teaches geometric mapping of threads to processors based on cache locality.

Alfieri (USPN 5,745,778) teaches global scheduling and load balancing using processor affinity to improve performance.

Neufeld (USPN 5,974,438) teaches using a scoreboard method to track a thread's cache usage.

Sundaresan (USPN 6,289,369) teaches a combined affinity and load balancing method for threads using cache locality information.

Bellosa et al. ("The Performance Implications of Locality Information Usage in Shared-Memory Multiprocessors"), Sinharoy ("Optimized Thread Creation for Processor Multithreading"), Nikolopoulos et al. ("Efficient Runtime Thread Management for the Nan-Threads Programming Model"), and Weissman ("Performance Counters and State Sharing Annotations: a Unified Approach to Thread Locality") are referenced by Applicant in the specification and are generally related to scheduling methods using thread locality.

43. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Syed Ali
April 21, 2004

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100